AFRL-AFOSR-JP-TR-2016-0058

Prediction of User Context Using Smartphone Activity Data

Jae-Gil Lee
Korea Advanced Institute of Science and Technology

07/22/2016
Final Report

Air Force Research Laboratory
AF Office Of Scientific Research (AFOSR)/ IOA
Arlington, Virginia 22203
Air Force Materiel Command

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
22-07-2016

2. REPORT TYPE
Final

3. DATES COVERED *(From - To)*
28 May 2015 to 27 May 2016

4. TITLE AND SUBTITLE
Prediction of User Context Using Smartphone Activity Data

5a. CONTRACT NUMBER

5b. GRANT NUMBER
FA2386-15-1-4060

5c. PROGRAM ELEMENT NUMBER
61102F

6. AUTHOR(S)
Jae-Gil Lee

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Korea Advanced Institute of Science and Technology
291 Daehak-ro, Yuseong-gu
Taejon, 305701 KR

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AOARD
UNIT 45002
APO AP 96338-5002

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/AFOSR IOA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-AFOSR-JP-TR-2016-0058

12. DISTRIBUTION/AVAILABILITY STATEMENT
A DISTRIBUTION UNLIMITED: PB Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT
When a person seeks another persons attention, it is of prime importance to assess how interruptible the other person is. Since smartphones are ubiquitously used as communication media these days, interruptibility prediction on smartphones has started to attract great interest from both academia and industry. Previous studies, in general, attempted to model interruptibility using the behaviors at the current moment and in the immediate past (e.g., 5 minutes before). However, a persons interruptibility at a certain moment is indeed affected by his/her preceding behaviors for several reasons. Motivated by this long-term effect, in this project we propose a novel methodology of extracting features based on past behaviors from smartphone sensor data. The primary difference from previous studies is that we systematically consider a longer history of up to a day in addition to the current point and the immediate past. To represent behaviors in a day accurately and compactly, our methodology divides a day into multiple timeslots and then, for each timeslot, derives relevant features such as the temporal shapes of the time series of the sensor data. In order to verify the advantage of our methodology, we collected a data set of smartphone usage from 25 participants for four weeks and obtained a license to a large-scale public data set constructed from 907 users over approximately 9 months. The experimental results on the two data sets show that looking back to the beginning of the current day improves prediction accuracy by up to 13% and 8%, respectively, compared with the baseline and state-of-the-art methods.

15. SUBJECT TERMS
Smartphone

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
Unclassified	Unclassified	Unclassified	SAR	21	KNOPP, JEREMY

19b. TELEPHONE NUMBER *(Include area code)*
315-227-7006

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

Final Report for AOARD Grant FA2386-15-1-4060

Prediction of User Context Using Smartphone Activity Data

5/27/2016

Jae-Gil Lee:

- e-mail address : jaegil@kaist.ac.kr
- Institution : Korea Advanced Institute of Science and Technology
- Mailing Address : 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea
- Phone : +82-42-350-1617
- Fax : +82-42-350-1610

Period of Performance: 5/28/2015 – 5/27/2016

ABSTRACT

When a person seeks another person's attention, it is of prime importance to assess how *interruptible* the other person is. Since smartphones are ubiquitously used as communication media these days, *interruptibility prediction* on smartphones has started to attract great interest from both academia and industry. Previous studies, in general, attempted to model interruptibility using the behaviors at the current moment and in the immediate past (e.g., 5 minutes before). However, a person's interruptibility at a certain moment is indeed affected by his/her preceding behaviors for several reasons. Motivated by this long-term effect, in this project we propose a novel methodology of extracting features based on past behaviors from smartphone sensor data. The primary difference from previous studies is that we systematically consider *a longer history of up to a day* in addition to the current point and the immediate past. To represent behaviors in a day accurately and compactly, our methodology divides a day into multiple timeslots and then, for each timeslot, derives relevant features such as the temporal shapes of the time series of the sensor data. In order to verify the advantage of our methodology, we collected a data set of smartphone usage from 25 participants for four weeks and obtained a license to a large-scale public data set constructed from 907 users over approximately 9 months. The experimental results on the two data sets show that *looking back to the beginning of the current day* improves prediction accuracy by up to 13% and 8%, respectively, compared with the baseline and state-of-the-art methods.

INTRODUCTION

Human interruptibility, simply *interruptibility*, in general is defined by the degree of how opportune it is to interrupt a person [15]. The probability of replying to an instant message or checking a notification at a particular moment is a typical example of interruptibility. Then, interruptibility prediction is to assess another person's interruptibility prior to interaction with him/her [8, 14]. With accurate prediction, we can expect a quick and high-quality response to an interruption, and the cognitive burden of the person interrupted is reduced significantly [11]. Thus, the importance of interruptibility prediction is being widely recognized since it is beneficial for both those who interrupt and those who are interrupted [28]. Interruptibility prediction has been extensively studied in various scenarios: office environments [8, 14], desktop computers [12, 13], and mobile devices [18, 20, 21, 22, 23, 24, 27]. In particular, owing to the prevalence of mobile devices [4, 25, 27], huge amounts of research effort are currently being devoted to interruptibility prediction on mobile devices — smartphones. Previous studies have demonstrated that interruptibility can be predicted fairly well (with an accuracy of over 70%) using various types of context information. One of the active topics in this direction is to determine what data to capture to represent the current context [19, 28], because the advances in ubiquitous sensing technology provide us with abundant contextual data. Regardless of data sources, one dominating assumption is that the current context can

best be modeled by the observations obtained at that exact point in time. That is, previous studies attempted to represent interruptibility mostly using the “present-time” features captured at the specific moment. Examples of these features include the current ringer mode and the current screen on/off status. In addition to these features, more recent studies have started considering users’ past behaviors, such as events occurring in the previous one or five minutes [8, 14, 24] and the time elapsed since the last event [20, 21, 22]. This is reasonable because the consequences of past behaviors and history are part of the current context [9]. Nonetheless, these studies still do not consider past behaviors extensively since they reflect only the immediate past.

In this research, we tackle the problem of **systematically incorporating past behaviors into interruptibility prediction**. Differing from existing research that considers only the present time and the immediate past, our methodology considers a longer history of up to one day. Here, we contend that proper consideration of past behaviors plays a key role in accurate prediction. The intuition behind our methodology is two-fold as shown below.

• **Self-regulation:** People consciously manage or guide their own thoughts and behaviors [16]. In addition, the amount of human activity per day is in fact limited and conserved [31]. Thus, for example, if a person did not concentrate on work in the morning, the person would probably work harder in the afternoon to finish a planned task within the day.
• **Prolongation:** The effect of an event could last for a long time [1]. Hence, for example, if a phone call to someone makes a caller feel relieved, the caller is more willing to do a favor after the phone call during the entire day.

Improving interruptibility prediction based on past behaviors is challenging. First, it is important to determine how far back we need to look. We empirically verify that looking back on the current day is sufficient to achieve the best result. Furthermore, since a temporal window is relatively long (i.e., from several hours to a day), a novel approach to feature extraction from smartphone usage data is needed for effective prediction. We carefully derive relevant features that include the statistical measures, value distributions, and temporal shapes of the time series of smartphone sensor data. Figure 1 shows the concept of our methodology.

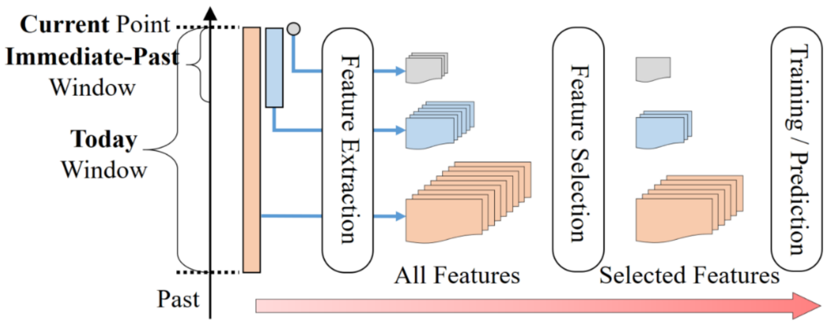


Figure 1. The main concept of our methodology.

In addition to the features extracted from the present time and the immediate past, those extracted *from the current moment back to the beginning of a day* are provided to the feature selection module. Many more features are derived from the today window than from the current point and the immediate-past window because of its longer duration. Finally, only the discriminative features resulting from the feature selection module are used for training and prediction. We note that our work is orthogonal to existing work that explores predictive data sources (e.g., [17]). Our sophisticated design supports any time-series data of numeric, binary, and nominal variables. Thus, given a set of attributes, we attempt to maximize the benefits of those attributes by harnessing daily behaviors.

RESEARCH QUESTIONS

Our research questions for this project are as follows.

- **RQ1:** Interruptibility is affected by the immediate past behavior as well as the current status.
- **RQ2:** The accuracy of interruptibility prediction improves significantly when using the behavior of the current day.
- **RQ3:** Looking further back beyond the current day is not very helpful for interruptibility prediction.
- **RQ4:** The behavior of the target day can be replaced with that of a preceding day without reducing much accuracy.

INTERRUPTIBILITY DATA SETS

We used two real-world smartphone usage data sets: the KAIST data set and the Device Analyzer data set [29]. The former is our proprietary data set, and the latter is a public data set. Especially, the Device Analyzer data set is known to be the largest collection of smartphone usage data, and we extracted 907 users for use in our experiment in the order of the number of instances recorded. In both data sets, hour of day and day of week were attached to every recording. Interruptibility is modeled as a binary state as typically done by recent studies [18, 21, 22, 24]. Table 1 shows the general statistics of the two data sets. The first column represents the number of attributes which will be detailed in Tables 2 and 3 respectively. The total number of interruptibility labels (interruptible or non-interruptible) is reported in the last column.

Table 1. Statistics of the two data sets.

Data Set	# Attributes.	# Users	# Labels
KAIST	24	25	4,103
Device Analyzer	26	907	1,870,315

KAIST Data Set

Participants

We conducted a field study with 25 participants who installed our own data-collection application and reported their data for four weeks. The goal of this study was to obtain not only smartphone usage data but also the ground truth on the participants' interruptibility through experience sampling. Among 25 participants, 5 were recruited from our department, and 20 were from an online community. For the former group, we personally asked them to join the experiment; for the latter group, we posted a wanted advertisement on the online community and selected 20 eager users from the applicants. Then, we provided all participants with the detailed instructions and received explicit consent from them before the experiment. After the experiment was complete, we paid about US$100 to each participant. This study was approved by the KAIST institutional review board (IRB).

Data Collection

All participants downloaded the data-collection application from Google Play and installed it on their Android smartphone. Our application supports Android 4.0 or higher. Table 2 shows the attributes that the application collected in the background. Each participant sent us his/her weekly data at the end of each week, and we verified the data to give him/her feedback on the quality of the data. We did not collect any personal information that can be used for inferring the data owner as per the recommendation of the IRB. This data collection was run for four weeks from February 2015 to March 2015.

Table 2. Attributes collected in the KAIST data set.

Attribute	Description	Type	# Instances
cpu	CPU usage	numeric	67,590
bat_lev	Battery level	numeric	242,560
bat_temp	Battery temperature	numeric	242,560
cell_strn	Cellular signal strength	numeric	95,071
wifi_strn	WiFi signal strength	numeric	40,482
ill	Ambient light level	numeric	37,093
accel_x	Acceleration force (X-axis)	numeric	119,347
accel_y	Acceleration force (Y-axis)	numeric	119,347
accel_z	Acceleration force (Z-axis)	numeric	119,347
accel_tot	Acceleration force (total)	numeric	119,347
airplane	Airplane mode on/off	binary	67,590
screen	Screen on/off	binary	67,590
headset	Headset mode on/off	binary	67,590
cell	Cellular mode on/off	binary	95,071
wifi	WiFi mode on/off	binary	40,482
charge	Charge mode on/off	binary	242,560
ringtone	Ringtone mode	nominal	67,590
charge_stat	Charge status	nominal	242,560
ssid	Connected WiFi SSID	nominal	40,482
app_pkg	Application package name	nominal	264,520
app_cat	Application category	nominal	264,520
location	Location name (district)	nominal	52,744
call	Phone call event	nominal	3,530
sms	Message event	nominal	4,964

Experience Sampling

We collected the ground-truth information about the participants' state of interruptibility via experience sampling [5]. The experience sampling method (ESM) is a signal-contingent method of data collection from participants about their current experience or situation. In our case, we collected in-situ self-reports on the subjective state of interruptibility.

A notification in Figure 2 popped up — five times a day randomly between 9 a.m. and 10 p.m. — per trigger from our server. The notification asked the participants to answer the question with "Yes" or "No". All participants were explained about the meaning of the question: "you are interruptible if you are willing to do a simple task by spending less than ten minutes right now."

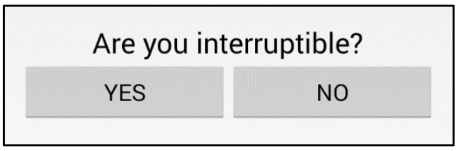


Figure 2. Screen capture of the experience sampling probe.

A participant's interruptibility was recorded together with temporal information (e.g., time and date) when he/she responded to a question. If a participant did not respond within ten minutes after receiving a question, we recorded his/her status as "not interruptible" at that time.

Device Analyzer Data Set

Description

The Device Analyzer project is being maintained by the University of Cambridge[1]. Its data set contains over 100 billion records of Android smartphone usage from over 17,000 devices across the world, which is known to be the largest collection of smartphone usage data. We officially obtained a license from the University of Cambridge and downloaded the snapshot of the data set as of November 2015. The size of the raw data reached around 7.5 terabytes. While the project collects more than 50 attributes, we selected the 26 attributes that correspond to those of the KAIST data set. Table 3 lists all the attributes used in this project.

Table 3. Attributes used in the Device Analyzer data set.

Attribute	Description	Type	# Instances
bat_lev	Battery level	numeric	180,018,744
bat_temp	Battery temperature	numeric	180,018,744
vol_music	Media (music) volume	numeric	56,860,086
vol_alarm	Alarm sound volume	numeric	56,860,086
vol_voicecall	Voice call sound volume	numeric	56,860,086
vol_system	System sound volume	numeric	56,860,086
vol_ring	Ringtone sound volume	numeric	56,860,086
vol_noti	Notification sound volume	numeric	56,860,086
accel	Acceleration force	numeric	29,118,563
light	Ambient light level	numeric	23,184,758
sms_unread_cnt	Number of unread SMS	numeric	4,176,962
airplane	Airplane mode on/off	binary	8,079,408
screen	Screen on/off	binary	36,064,991
headset	Headset mode on/off	binary	1,526,204
wifi	WiFi mode on/off	binary	2,028,945
wifi_conn	WiFi connectivity	binary	4,483,934
mobile_conn	Mobile connectivity	binary	6,290,799
bluetooth	Bluetooth on/off	binary	229,058
charge	Charge mode on/off	binary	9,695,334
ringtone	Ringtone mode	nominal	8,368,493
charge_stat	Charge mode on/off	nominal	9,695,334
display_orient	Display orientation	nominal	9,403,893
app_pkg	Application package name	nominal	92,774,207
app_cat	Application category	nominal	92,774,207
location	Location (LAC, CID)	nominal	60,754,805
sms	Message event	nominal	1,334,568

Among 9,641 users in total, we extracted the users who had sufficiently many records to achieve reliable results. Figure 3 shows the distribution of the number of incoming calls for each user, which follows a power-law distribution. We calculated the 70th percentile in the total numbers of calls and chose the users who had more calls than the 70th percentile. In this way, we extracted 907 heavy users from the entire set of users, as indicated by the yellow area in Figure 3. They contributed 70% of all incoming calls. In addition, their data were recorded for 274 days on the average.

[1] https://deviceanalyzer.cl.cam.ac.uk/

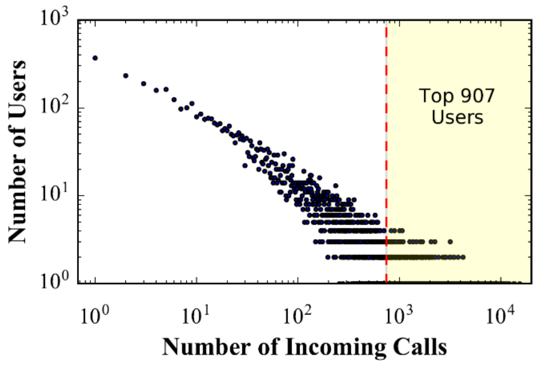


Figure 3. Distribution of the number of incoming calls per user in the Device Analyzer data set.

Ground Truth

Since the Device Analyzer data set does not contain experience sampling data, we treat call availability [20] as interruptibility. Alternative to the ESM, this implicit labeling involves observing user actions and making deductions, just like Pielot et al. [21] did using notification dismissal. Our labeling is reasonable because users in non-interruptible circumstances cannot or do not pick up incoming calls because of unavoidable, enforced, intentional, or negligent unavailability [23]. We excluded call-related attributes from prediction since they directly indicate interruptibility. A user is regarded as being interruptible when he/she picks up an incoming call and continues the call for at least ten seconds. In contrast, a user is regarded as being not interruptible when he/she does not pick up an incoming call or quits the call within just ten seconds.

Detailed Statistics

In order to examine the data sets at fine granularity, we divide a day into six equi-width timeslots as in Figure 4.

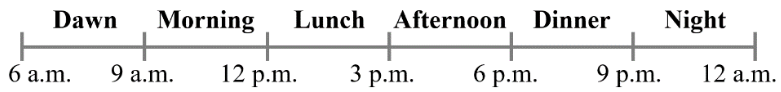


Figure 4. Six timeslots in a day.

Figure 5 shows the number of interruptibility labels (i.e., interruptible or non-interruptible) per user in each timeslot through the experimental period. The median numbers range between 32 and 47 in Figure 5(a) and between 260 and 298 in Figure 5(b). While the KAIST data set has a sufficient number of labels, the Device Analyzer data set has a significantly larger number of labels.

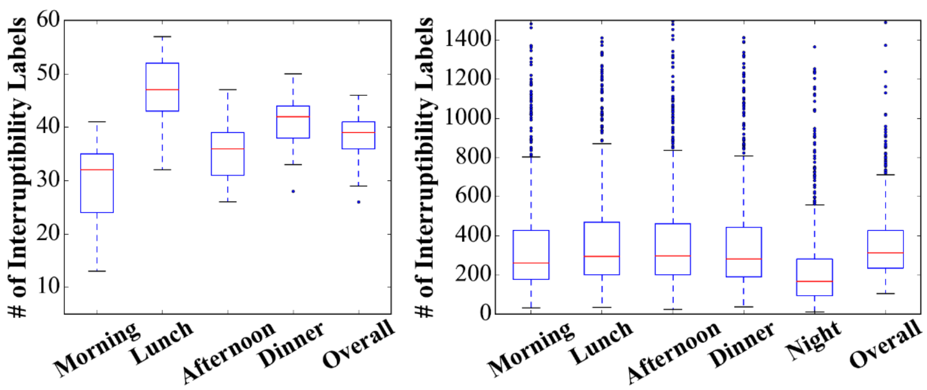


Figure 5. Number of interruptibility labels in each timeslot.

Figure 6 shows what proportion of the labels are interruptible or not in each timeslot. In Figure 6(a), 41.2% of the labels indicate being interruptible, and 58.8% of the labels indicate being not interruptible. In Figure 6(b), the corresponding proportions are 55% and 45% respectively. The proportion of being interruptible is higher in Figure 6(b) than in Figure 6(a), because talking over the phone is easier than accepting a simple task. We note that the two label values tend to be balanced across all timeslots in both data sets.

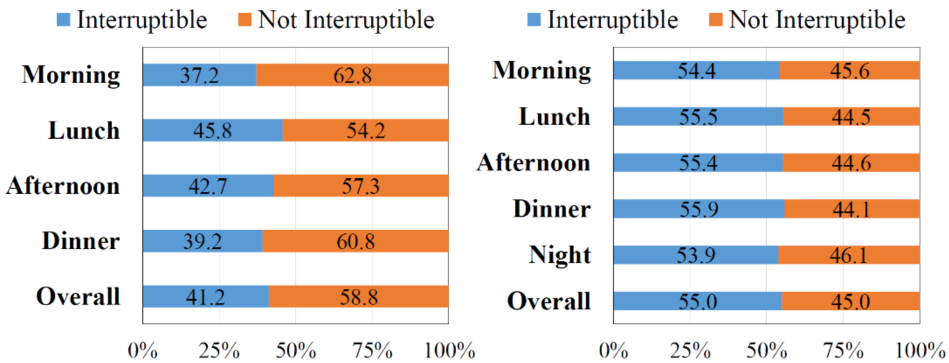


Figure 6. Proportion of interruptibility label values in each timeslot.

METHODOLOGY

In this section, we propose our methodology of extracting daily features and modeling the interruptibility using the extracted features

Temporal Windows

First of all, in order to answer RQ1–RQ4, we define three types of temporal windows in Figure 7 and consider them together with the current point. In Definition 1, we clarify the source of a feature depending on whether it is extracted from the current point or a temporal window.

- **Current point:** the current moment when interruptibility needs to be predicted
- **Immediate-past window:** the interval from the current point back to 15 minutes before
- **Today window:** the interval from the current point back to the beginning of the current day
- **Yesterday window** (or **the-day-before-yesterday window**)**:** the interval from the end of the latest previous day (or the second-latest previous day) back to the beginning of the latest previous day (or the second-latest previous day)

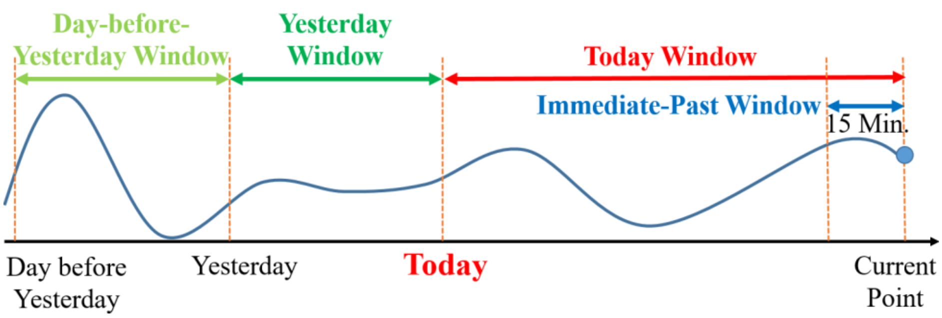


Figure 7. Temporal windows used for feature extraction.

We define the *basic* features as those extracted from the current point in Figure 7, and the *extended* features as those extracted from a temporal window in Figure 7. We specifically call the extended features from the today window the *daily* features.

Extended Features

To cover various data sources, we categorize attributes (variables) into three types: numeric, binary, and nominal attributes, as shown in Figure 8. Since each attribute type has distinct characteristics, we define the extended features separately for each type so that they best represent the attribute values of the type in a given temporal window.

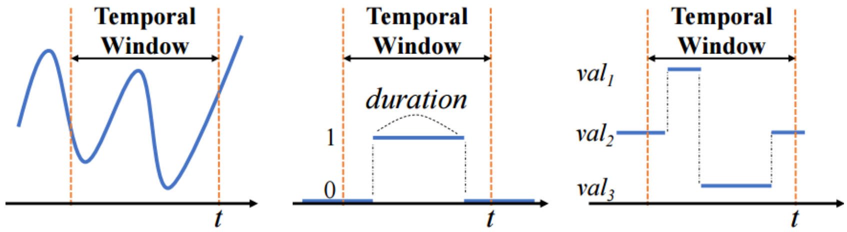


Figure 8. Three types of the attributes in the interruptibility data sets.

Overview

Table 4 shows the list of extended features for each attribute type, complementary to Figure 8. For numeric attributes, the mean and standard deviation are calculated to represent the central tendency and dispersion of the values in a temporal window; in addition, a discrete wavelet transform (DWT) is applied to capture the general trend (i.e., shape) in a given window, which will be discussed in detail. For (asymmetric) binary attributes, since a positive value is more important, we keep the duration of "1" samples and the number of transitions from "0" to "1" in a temporal window. Since a nominal attribute is a generalization of a binary attribute, we keep such duration and number for each possible value.

Table 4. Extended features derived from a temporal window.

Measure	Description
Numeric Attributes (Figure 8(a))	
mean	the mean of the samples
std	the standard deviation of the samples
dwt	the 32 DWT coefficients of the samples
Binary Attributes (Figure 8(b))	
dur	the sum of the duration of "1" samples
num	the total number of transitions to "1"
Nominal Attributes (Figure 8(c))	
$\mathrm{val_i}$_dur	the sum of the duration of "val_i" samples
$\mathrm{val_i}$_num	the total number of transitions to "val_i"

While the current point and the immediate-past window are considered as atomic units, the today window, the yesterday window, and the day-before-yesterday window are partitioned into six timeslots — dawn, morning, lunch, afternoon, dinner, and night — according to Figure 4 before deriving extended features. For the today window, the timeslot to which the current point belongs is considered partially up to the present time. For example, if the present time is 8 p.m., the timeslot dinner spans from 6 p.m. to 8 p.m. (not 9 p.m.). The goal of this partition is to shorten the length of a temporal window such that each interval has coherent semantics, because considering a too long interval as a whole may lose important information

We now summarize how an extended feature is constructed in Figure 9. If a temporal window is immediate-past, the measures except *dwt* in Table 4 are calculated for the window. Otherwise, a window is split into timeslots, and then the measures are calculated for each timeslot. A feature name is denoted by concatenating the names of an attribute, a temporal window, a timeslot, and a measure, e.g., *cpu_today_lunch_std*.

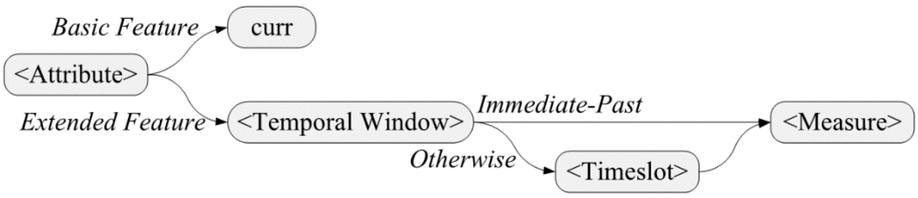


Figure 9. Notation and composition of extended features.

Discrete Wavelet Transform (DWT) Features

The DWT has been widely used for compression and dimensionality reduction owing to its capability of capturing the major trends of underlying data. It decomposes an input sequence into a set of wavelets and produces a set of coefficients of the same size as the sequence. The Haar wavelet [2], which is very simple yet effective, is adopted in this work. The coefficients, as going from the first to the last, indicate the frequency of a finer temporal domain. A key advantage over other transforms (e.g., Fourier transforms) is temporal resolution that captures location in time as well as frequency, which is essential for our problem since the time when an event happened should be preserved. Before going into the details, we present a motivating example for the DWT.

Motivation Example: Figure 10 shows the values of accel_x in the KAIST data set. Between the two different timeslots 6 of Figures 10(a) and 10(b), the mean, which is denoted by the red dashed line, and the standard deviation are almost the same, although the shapes are very different from each other. User 22 almost did not move on February 14 (Figure 10(a)), whereas User 22 frequently moved on February 24 (Figure 10(b)). In fact, the interruptibility at the dinner on February 14 was different from that at the dinner on February 24. Thus, the temporal shape is related to interruptibility, and the DWT is needed to capture the temporal shape that can be characterized neither by the mean nor by the standard deviation.

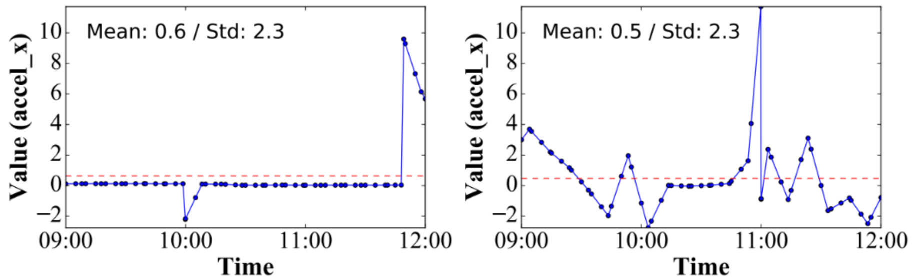


Figure 10. A motivating example for using the DWT.

We derive DWT coefficients for a sequence generated from each 3-hour timeslot covered by a temporal window. We do not apply the DWT to the immediate-past window since its duration is not long enough. First, a sequence of length 180 is constructed by the values at every minute. If a value does not exist, it is estimated by linear interpolation between two consecutive timestamps. Then, we pad zeros on the right side of the sequence to make its length 256 because the DWT is defined for the sequences with length of a power of 2. Last, after applying the DWT to the input sequence, only the first 32 coefficients are selected for dimensionality reduction, and such an approach is widely accepted when leveraging DWT coefficients as an attribute [3, 30]. Figure 11 shows the sequences obtained by restoring those sequences in Figure 10 with the 32 coefficients. For both sequences in Figures 11(a) and 11(b), the shape of the restored sequence in red is very close to that of the original sequence in blue.

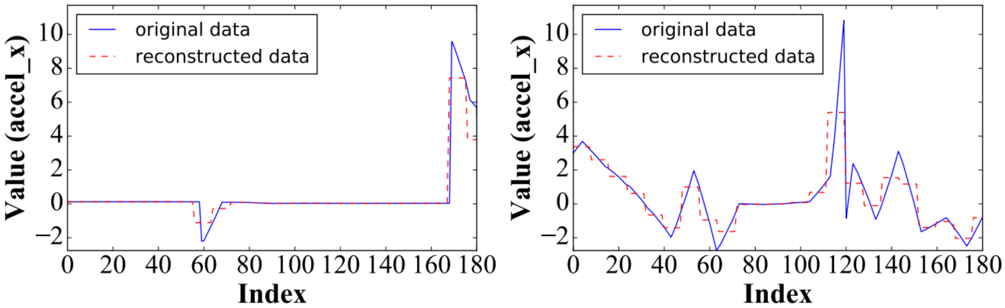


Figure 11. Reconstruction using the first 32 Haar wavelet coefficients.

Feature Configurations

Table 5 illustrates the definitions of the seven feature configurations subsequently used in this project. CURR takes account of the current point only. IPAST expands feature extraction to the immediate past. DAY[] takes advantage of the features constructed from a long duration in addition to those used by IPAST; 0 indicates the current day, -1 one day before that day, and -2 two days before that day; a colon denotes an inclusive range. For example, DAY[0] takes account of the current point, the immediate-past window, and the today window.

Table 5. Feature configurations used in this project.

Time / Conf.	D-b-Yesterday	Yesterday	Today	Imm-Past	Current
Curr					
IPAST					
DAY[0]					
DAY[-1:0]					
DAY[-2:0]					
DAY[-1]					
DAY]-2]					

For example, if we consider a numeric attribute *cpu* and current point appears in the timeslot *night*, the configuration DAY[0] produces 21 features in total, as shown in Figure 12. Concatenation of the nodes by following arrows from the root to a leaf composes a feature.

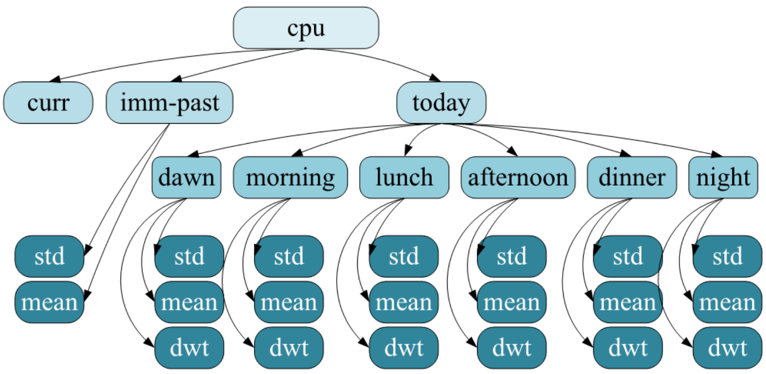


Figure 12. List of all features for the attribute cpu in DAY[0].

EVALUATION

In this section, we report the results of a series of experiments designed to answer each research question.

Experimental Setting

Data Preprocessing

We improved the quality of the raw data by preprocessing. First, numeric attribute values were normalized to between 0 and 1 by min-max normalization and then discretized by the MLDPC method [7] that determines the optimal cut points by supervised learning. Second, the instance timestamps were made the same across all attributes of a user. A missing value at a certain timestamp was estimated by linear interpolation for numeric attributes and by forward filling, which uses the immediately previous value, for binary and nominal attributes. Third, if there were too many possible values in a nominal attribute (e.g., location identifiers and application names), we selected the most frequent 10 values and grouped all other infrequent values into a single value.

Feature Selection and Prediction

Regarding feature selection, we used the correlation-based feature selection (CFS) [10] method implemented in Weka. The CFS method selects a subset of features that are highly correlated with the class while having low intercorrelation. Regarding prediction, we used four classification methods: naive Bayes classifier (**NB**), support vector machine (**SVM**), random forest (**RF**), and C4.5 decision tree (**C4.5**). We present only the results of naive Bayes classifiers because of its highest accuracy.

Compared Methods

We compared the seven feature configurations in Table 5.

- **Baseline (CURR):** corresponding to earlier work (e.g., [15]) that uses only the present-time features
- **State-of-the-art (IPAST):** corresponding to recent work (e.g., [8, 14, 24, 25]) that uses the immediate-past features as well
- **Proposed methodology (DAY[0]):** using the daily features as well
- **Variation (DAY[-1:0], DAY[-2:0], DAY[-1], DAY[-2]):** using the data of one or two days ago

Data Sets

We used the data on all days for RQ1 and RQ2, but we used the data only on Wednesday, Thursday, and Friday for RQ3 and RQ4. When we address RQ3 and RQ4, since the yesterday and the-day-before-yesterday windows are additionally considered, we want to make sure that all temporal windows span through weekdays in order to avoid a possible bias between weekdays and weekends. In addition, the timeslot *night* was not used for prediction in the KAIST data set owing to the lack of the ground truth, whereas it was used in the Device Analyzer data set.

Table 6 shows the total number of features extracted by each configuration when the current point belongs to dinner. Only the number of DAY[0] is affected by the current point since it includes the timeslots up until that point, whereas those of the other configurations are not. The number of features increase as the duration used for feature extraction gets longer.

Table 6. Number of features used for prediction in dinner.

Data Set / Conf.	KAIST (Dinner)	Device Analyzer (dinner)
CURR	70	71
IPAST	195	199
DAY[0]	2,420	2,599
DAY[-1], DAY[-2]	2,865	3,079
DAY[-1:0]	5,090	5,479
DAY[-2:0]	7,760	8,359

Measurement

We built a personalized classification (prediction) model for each person using his/her own data only. Then, we measured the accuracy in Eq. (1) by 5-fold cross validation for the relatively small KAIST data set and 10-fold cross validation for the Device Analyzer data set. In Eq. (1), TP, FP, FN, and TN indicate the numbers of true positives, false positives, false negatives, and true negatives, respectively. Last, we reported the average of the accuracy values from all users.

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \qquad (1)$$

The significance of the difference between accuracy values was tested for all pairs of configurations using the two-tailed t-test. The t-test was performed on overall accuracy, i.e., the average of the accuracy values on all timeslots. The results are summarized in Tables 7 and 8, where the p-value and the number (N) of samples used for each test are presented. The number of asterisks denotes the statistical significance. The t-test results that correspond to RQ1, RQ2, RQ3, and RQ4 are indicated in colors of orange, yellow, green, and blue, respectively.

Table 7. T-test results for the KAIST data set.

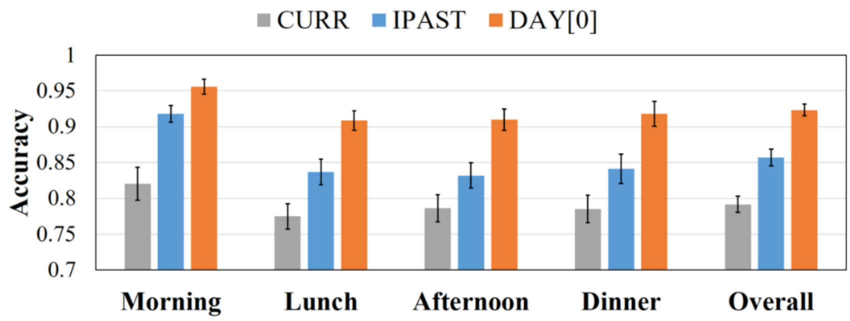
NS: $p \geq 0.05$, *: $p<0.05$, **: $p<0.01$, ***: $p<0.001$, ****: $p<0.0001$

		CURR	IPAST	DAY[0]	DAY[-1:0]	DAY[-2:0]	DAY[-1]	DAY[-2]
CURR	P-value							
	N							
IPAST	P-value	0.0002***						
	N	RQ1 25						
DAY[0]	P-value	0****	0****					
	N	RQ2 25	25					
DAY[-1:0]	P-value	0****	0.0109*	NS				
	N	24	24	24				
DAY[-2:0]	P-value	0****	0.0109*	NS	NS			
	N	24	24	RQ3 24	24			
DAY[-1]	P-value	0****	NS	0.018*	0.018*	0.018*		
	N	24	24	24	24	24		
DAY[-2]	P-value	0****	NS	0.018*	0.018*	0.018*	NS	
	N	24	24	RQ4 24	24	24	24	

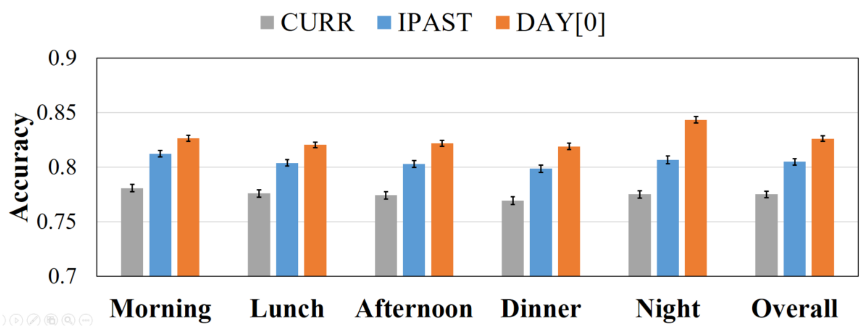
Table 8. T-test results for the Device Analyzer data set.
NS: $p \geq 0.05$, *: $p<0.05$, **: $p<0.01$, ***: $p<0.001$, ****: $p<0.0001$

		CURR	IPAST	DAY[0]	DAY[-1:0]	DAY[-2:0]	DAY[-1]	DAY[-2]
CURR	P-value							
	N							
IPAST	P-value	0****						
	N	RQ1 907						
DAY[0]	P-value	0****	0****					
	N	RQ2 907	907					
DAY[-1:0]	P-value	0****	0****	0.0487*				
	N	883	883	883				
DAY[-2:0]	P-value	0****	0****	0.008**	NS			
	N	883	883	RQ3 883	883			
DAY[-1]	P-value	0****	0****	0.0128*	0****	0****		
	N	883	883	883	883	883		
DAY[-2]	P-value	0****	0****	0.0212*	0****	0***	NS	
	N	883	883	RQ4 883	883	883	883	

RQ1 and RQ2: Daily Features
Figure 13 shows the accuracy calculated based on different features to address RQ1 and RQ2 for both data sets. Error bars indicate the standard error.



(a) KAIST data set.



(b) Device Analyzer data set

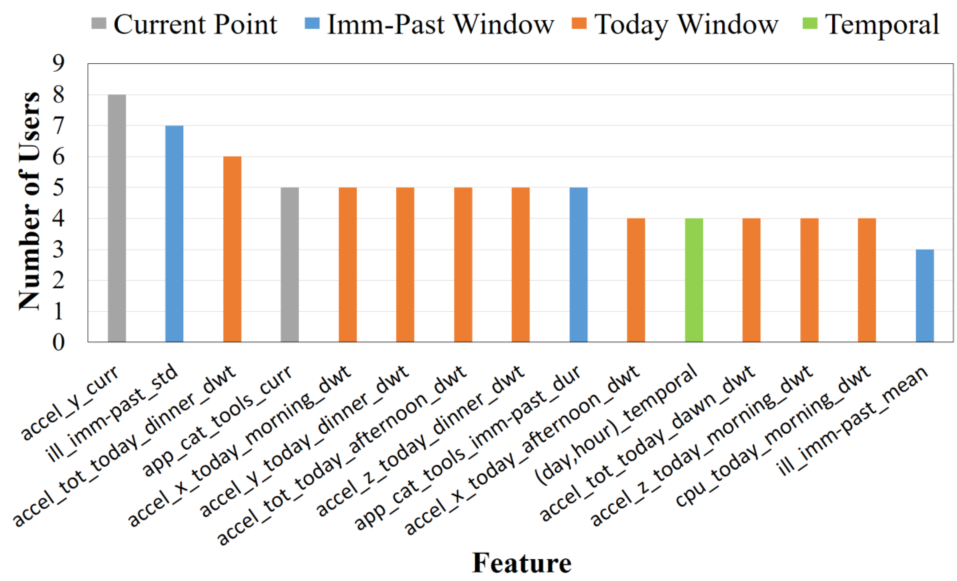
Figure 13. Accuracy based on different features to address RQ1 and RQ2

In both data sets, DAY[0] achieved the highest accuracy, followed by IPAST and CURR, for all timeslots. In addition, as the orange and yellow areas in Tables 7 and 8 which correspond to RQ1 and RQ2 respectively show, there are statistically significant differences between CURR and IPAST and between IPAST and DAY[0]. Table 9 shows the results of all four classifiers for the experiment in Figure 13, where the colored cells correspond to the values in the plot. Here, there was no big difference among the classifiers.

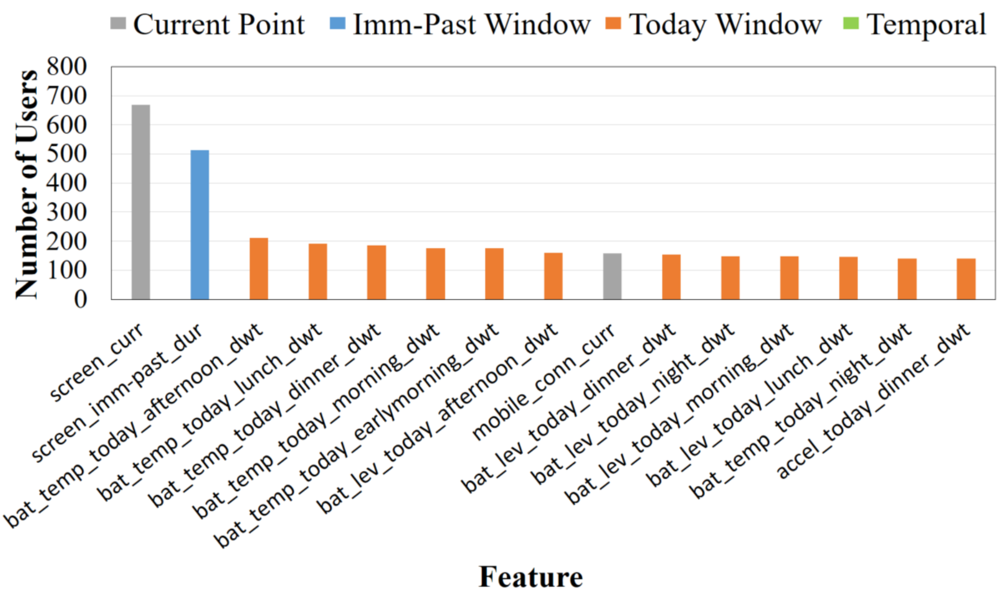
Table 9. Accuracy results for RQ1 and RQ2 with all four classifiers.

Conf.	Accuracy (KAIST)				Accuracy (Device Analyzer)			
	NB	SVM	RF	C4.5	NB	SVM	RF	C4.5
Morning								
CURR	0.82	0.75	0.76	0.77	0.78	0.77	0.77	0.77
IPAST	0.91	0.86	0.87	0.84	0.82	0.80	0.81	0.80
DAY[0]	0.95	0.91	0.91	0.84	0.83	0.82	0.82	0.82
Lunch								
CURR	0.77	0.71	0.70	0.74	0.78	0.76	0.77	0.77
IPAST	0.83	0.79	0.79	0.79	0.81	0.79	0.80	0.80
DAY[0]	0.90	0.87	0.86	0.82	0.83	0.81	0.82	0.81
Afternoon								
CURR	0.78	0.73	0.72	0.74	0.78	0.76	0.77	0.77
IPAST	0.83	0.81	0.7	0.78	0.81	0.79	0.80	0.80
DAY[0]	0.91	0.87	0.88	0.83	0.83	0.81	0.82	0.81
Dinner								
CURR	0.78	0.73	0.71	0.75	0.77	0.76	0.76	0.76
IPAST	0.84	0.79	0.79	0.79	0.80	0.79	0.80	0.79
DAY[0]	0.91	0.89	0.89	0.84	0.83	0.81	0.82	0.81
Night								
CURR	-	-	-	-	0.78	0.76	0.76	0.76
IPAST	-	-	-	-	0.81	0.79	0.80	0.79
DAY[0]	-	-	-	-	0.85	0.84	0.84	0.82
Overall								
CURR	0.79	0.73	0.73	0.75	0.78	0.76	0.77	0.77
IPAST	0.85	0.81	0.81	0.80	0.81	0.79	0.80	0.80
DAY[0]	0.92	0.89	0.88	0.83	0.84	0.82	0.82	0.81

Figure 14 shows the top-15 discriminative features for DAY[0]. Here, the importance of a feature is determined by the number of users whose model still contains it after feature selection. We show the results only for the last timeslot — dinner for the KAIST data set and night for the Device Analyzer data set — to avoid redundancy since we found a persistent consistency among all timeslots. As shown in Figure 14, the features from the current point and the immediate-past window were ranked the first and the second respectively. However, the other features were mostly extracted from the today window. Interestingly, even though we predicted the interruptibility for the last timeslot, many of these "today-window" features came from earlier timeslots (even including dawn): 6 out of 9 in the KAIST data set and 10 out of 12 in the Device Analyzer data set. This indeed confirms our claim that the behaviors in the previous several hours affect the current interruptibility.



(a) KAIST data set (for dinner).



(b) Device Analyzer data set (for night).

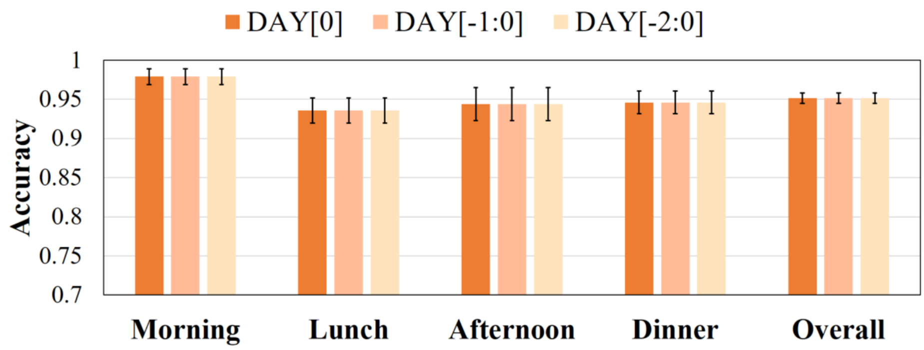
Figure 14. Top-15 discriminative features in DAY[0].

In Figure 14, we observe that many of the discriminative features for the KAIST data set are accelerometer-related ones and those for the Device Analyzer data set are screen or battery-related ones, all of which are closely related to movement or usage of smartphones. This result on important sensor categories is consistent with Dey et al. [6]'s work.

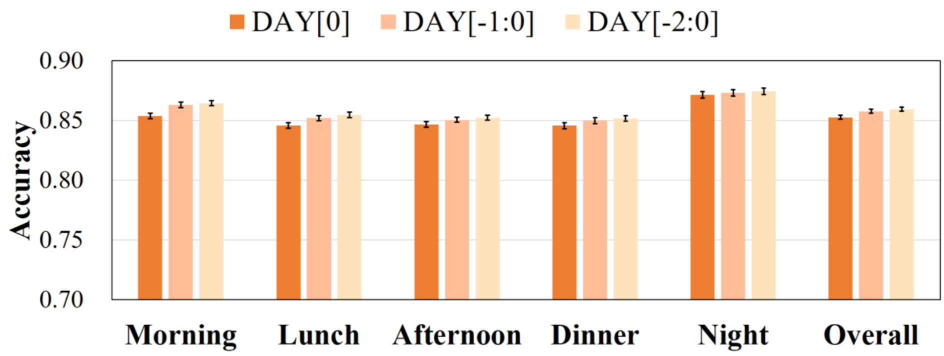
In conclusion, although the accuracy achieved by using only the basic features — 79% in the KAIST data set and 77% in the Device Analyzer data set in overall — is also acceptable, we can even increase the accuracy by leveraging the daily features.

RQ3: Temporal Window Length

Figure 15 shows the accuracy calculated based on different features to address RQ3 for both data sets.



(a) KAIST data set.



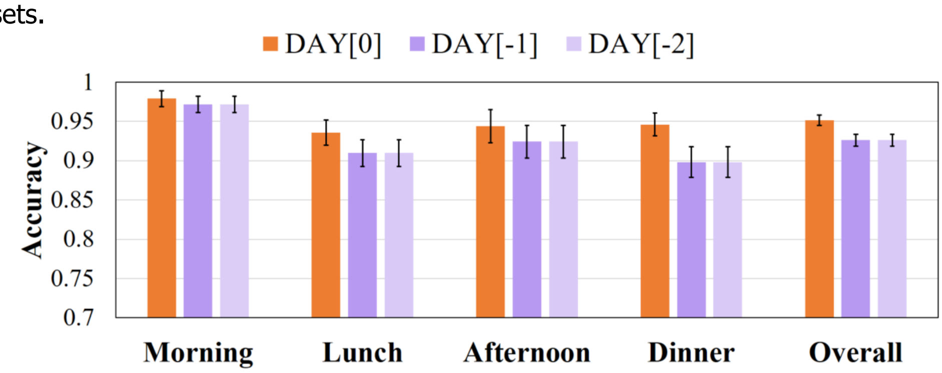
(b) Device Analyzer data set.

Figure 15. Accuracy based on different features to address RQ3.

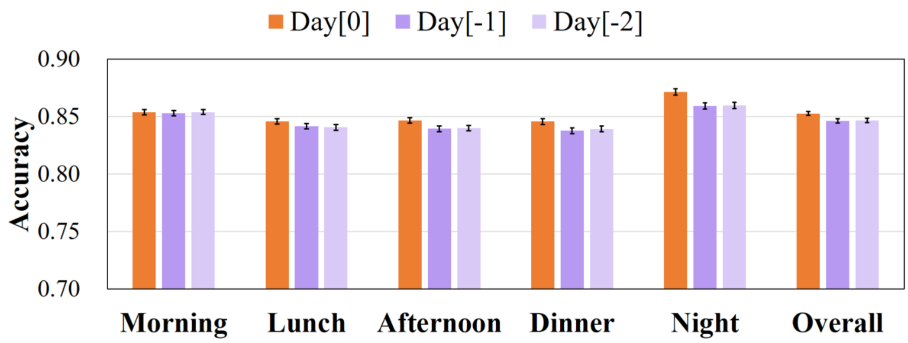
Prior to feature selection, DAY[-1:0] and DAY[-2:0] produce more (about twice or three times) features than DAY[0] because additional timeslots are considered in the yesterday and the-day-before-yesterday windows, as shown in Table 6. Despite a larger number of features, however, we did not observe significant increases in accuracy for DAY[-1:0] and DAY[-2:0] compared with DAY[0]. In particular, there was almost no increase in accuracy in the KAIST data set on all timeslots. On the other hand, in the Device Analyzer data set, the accuracy for DAY[-1:0] or DAY[-2:0] was slightly higher than that for DAY[0]. However, this increase is not statistically significant at the significance level of 0.01, as shown in Table 8. In conclusion, looking further back beyond the current day is not very helpful for increasing the prediction accuracy of interruptibility.

RQ4: Daily Routineness

Figure 16 shows the accuracy calculated based on different features to address RQ4 for both data sets.



(a) KAIST data set.



(b) Device Analyzer data set.

Figure 16. Accuracy based on different features to address RQ4.

It was observed that the accuracy values for both DAY[-1] and DAY[-2] were slightly lower than that for DAY[0]. This implies that the data from the current day is more helpful to predict interruptibility than the data from the latest (or second-latest) previous day in spite of the repetitive daily patterns. However, the decrease in accuracy from DAY[0] to DAY[-1] or DAY[-2] is not statistically significant at the significance level of 0.01, as shown in Table 7. In conclusion, the data from the latest (or second-latest) previous day can be a good substitute when the model suffers from the lack of the data from the current day.

CONCLUSION

We proposed a feature extraction methodology for interruptibility prediction using smartphone usage data. We conducted a field study and performed extensive experiments on two real-world data sets. Our methodology of looking back on the current day achieved the accuracy of over 90%, being higher than the baseline and state-of-the art methods by up to 13% and 8% respectively. The improvement was attributed to the fact that daily behavioral features were included in the predictive features of many users. We also found out that looking further back beyond the current day did not improve accuracy owing to the daily routineness of human behaviors. We, thus, confirmed that a day's behavior is replaceable with another day's behavior for the same reason. We believe that smartphone applications benefiting from our methodology will improve communication efficiency dramatically, based on a better understanding of when and how to engage with users. A potential application scenario that we envision is the real-time mobile Q&A service. When a user asks a question on such a smartphone application, the question is delivered to a set of expert users; when some of them answer the question, the answers are immediately delivered to the questioner. Thus, the success of this service depends upon selection of expert users who are interruptible at that moment. While this work first proved the feasibility of exploiting daily features, as the future work we plan to further improve prediction accuracy by inventing new types of extended features.

REFERENCES

1. Christopher Beedie, Peter Terry, and Andrew Lane. 2005. Distinctions between Emotion and Mood. Cognition and Emotion 19, 6 (2005), 847–878.
2. C. Sidney Burrus, Ramesh A. Gopinath, and Haitao Guo. 1997. Introduction to Wavelets and Wavelet Transforms. Prentice Hall.
3. Pimwadee Chaovalit, Aryya Gangopadhyay, George Karabatis, and Zhiyuan Chen. 2011. Discrete Wavelet Transform-Based Time Series Analysis and Mining. Comput. Surveys 43, 2 (2011), 6:1–6:37.
4. Minsoo Choy, Jae-Gil Lee, Gahgene Gweon, and Daehoon Kim. 2014. Glaucus: Exploiting the Wisdom of Crowds for Location-Based Queries in Mobile Environments. In Proc. 8th Int'l Conf. on Weblogs and Social Media.
5. Thomas H. Davenport and John C. Beck. 2002. The Attention Economy: Understanding the New Currency of Business. Harvard Business Review Press.
6. Anind K. Dey, Katarzyna Wac, Denzil Ferreira, Kevin Tassini, Jin-Hyuk Hong, and Julian Ramos. 2011. Getting Closer: An Empirical Investigation of the Proximity of User to Their Smart Phones. In Proc. 2011 ACM Int'l Joint Conf. on Pervasive and Ubiquitous Computing. 163–172.
7. Usama M. Fayyad and Keki B. Irani. 1993. Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning. In Proc. 13th Int'l Joint Conf. on Artificial Intelligence. 1022–1027. 10
8. James Fogarty, Scott E. Hudson, Christopher G. Atkeson, Daniel Avrahami, Jodi Forlizzi, Sara Kiesler, Johnny C. Lee, and Jie Yang. 2005. Predicting Human Interruptibility with Sensors. ACM Trans. on Computer-Human Interaction 12, 1 (2005), 119–146.
9. Bernard Guerin. 2016. How to Rethink Psychology: New Metaphors for Understanding People and Their Behavior. Routledge.
10. Mark A. Hall. 1999. Correlation-based Feature Selection for Machine Learning. Ph.D. Dissertation. The University of Waikato, Hamilton, New Zealand.
11. Joyce Ho and Stephen S. Intille. 2004. Using Context-Aware Computing to Reduce the Perceived Burden of Interruptions from Mobile Devices. In Proc. 19th ACM Int'l Conf. on Computer-Supported Cooperative Work and Social Computing. 507–510.
12. Eric Horvitz and Johnson Apacible. 2003. Learning and Reasoning about Interruption. In Proc. 5th Int'l Conf. on Multimodal Interfaces. 20–27.
13. Eric Horvitz, Paul Koch, and Johnson Apacible. 2004. BusyBody: Creating and Fielding Personalized Models of the Cost of Interruption. In Proc. 19th ACM Int'l Conf. on Computer-Supported Cooperative Work and Social Computing. 507–510.
14. Scott Hudson, James Fogarty, Christopher Atkeson, Daniel Avrahami, Jodi Forlizzi, Sara Kiesler, Johnny Lee, and Jie Yang. 2003. Predicting Human Interruptibility with Sensors: A Wizard of Oz Feasibility Study. In Proc. 2003 SIGCHI Conf. on Human Factors in Computing Systems. 257–264.
15. Nicky Kern, Stavros Antifakos, Bernt Schiele, and Adrian Schwaninger. 2004. A Model for Human Interruptability: Experimental Evaluation and Automatic Estimation from Wearable Sensors. In Proc. 8th IEEE Int'l Sympo. on Wearable Computers. 158–165.
16. Traci Mann, Denise de Ridder, and Kentaro Fujita. 2013. Self-Regulation of Health Behavior: Social Psychological Approaches to Goal Setting and Goal Striving. Health Psychology 32, 5 (2013), 487–498.
17. Abhinav Mehrotra, Mirco Musolesi, Robert Hendley, and Veljko Pejovic. 2015. Designing Content-driven Intelligent Notification Mechanisms for Mobile Applications. In Proc. 2015 ACM Int'l Joint Conf. on Pervasive and Ubiquitous Computing. 813–824.
18. Veljko Pejovic and Mirco Musolesi. 2014. InterruptMe: Designing Intelligent Prompting Mechanisms for Pervasive Applications. In Proc. 2014 ACM Int'l Joint Conf. on Pervasive and Ubiquitous Computing. 897–908.
19. Veljko Pejovic and Mirco Musolesi. 2015. Anticipatory Mobile Computing: A Survey of the State of the Art and Research Challenges. Comput. Surveys 47, 3 (2015), 47:1–47:29.

20. Martin Pielot. 2014. Large-Scale Evaluation of Call-Availability Prediction. In Proc. 2014 ACM Int'l Joint Conf. on Pervasive and Ubiquitous Computing. 933–937.
21. Martin Pielot, Rodrigo de Oliveira, Haewoon Kwak, and Nuria Oliver. 2014. Didn't You See My Message?: Predicting Attentiveness to Mobile Instant Messages. In Proc. 2014 SIGCHI Conf. on Human Factors in Computing Systems. 3319–3328.
22. Martin Pielot, Tilman Dingler, Jose San Pedro, and Nuria Oliver. 2015. When Attention is not Scarce - Detecting Boredom from Mobile Phone Usage. In Proc. 2015 ACM Int'l Joint Conf. on Pervasive and Ubiquitous Computing. 825–836.
23. Antti Salovaara, Antti Lindqvist, Tero Hasu, and Jonna Hakkila. 2011. The Phone Rings But the User Doesn't Answer: Unavailability in Mobile Communication. In Proc. 13th Int'l Conf. on Human Computer Interaction with Mobile Devices and Services. 503–512.
24. Hillol Sarker, Moushumi Sharmin, Amin Ahsan Ali, Md. Mahbubur Rahman, Rummana Bari, Syed Monowar Hossain, and Santosh Kumar. 2014. Assessing the Availability of Users to Engage in Just-in-Time Intervention in the Natural Environment. In Proc. 2014 ACM Int'l Joint Conf. on Pervasive and Ubiquitous Computing. 909–920.
25. Choonsung Shin, Jin-Hyuk Hong, and Anind K. Dey. 2012. Understanding and Prediction of Mobile Application Usage for Smart Phones. In Proc. 2012 ACM Int'l Joint Conf. on Pervasive and Ubiquitous Computing. 173–182.
26. Chaoming Song, Zehui Qu, Nicholas Blumm, and Albert-László Barabási. 2010. Limits of Predictability in Human Mobility. Science 327, 5968 (2010), 1018–1021.
27. Vijay Srinivasan, Saeed Moghaddam, Abhishek Mukherji, Kiran K. Rachuri, Chenren Xu, and Emmanuel Munguia Tapia. 2014. MobileMiner: Mining Your Frequent Patterns on Your Phone. In Proc. 2014 ACM Int'l Joint Conf. on Pervasive and Ubiquitous Computing. 389–400.
28. Liam D. Turner, Stuart M. Allen, and Roger M. Whitaker. 2015. Interruptibility Prediction for Ubiquitous Systems: Conventions and New Directions from a Growing Field. In Proc. 2015 ACM Int'l Joint Conf. on Pervasive and Ubiquitous Computing. 801–812.
29. Daniel Wagner, Andrew Rice, and Alastair Beresford. 2013. Device Analyzer: Understanding Smartphone Usage. In Proc. 10th Int'l Conf. on Mobile and Ubiquitous Systems: Computing, Networking and Services. 195–208.
30. Yi-Leh Wu, Divyakant Agrawal, and Amr El Abbadi. 2000. A Comparison of DFT and DWT Based Similarity Search in Time-Series Databases. In Proc. 9th Int'l Conf. on Information and Knowledge Management. 488–495.
31. Kazuo Yano. 2014. Invisible Hand of Data. Soshisha.

List of Publications and Significant Collaborations that resulted from your AOARD supported project:

Minsoo Choy, Daehoon Kim, Jae-Gil Lee, Heeyoung Kim, and Hiroshi Motoda. 2016. Looking Back on the Current Day: Interruptibility Prediction Using Daily Behavioral Features. 2016 ACM Int'l Joint Conf. on Pervasive and Ubiquitous Computing. (under review)

Minsoo Choy, Daehoon Kim, Jae-Gil Lee, Heeyoung Kim, and Hiroshi Motoda. 2016. A Novel Framework for Interruptibility Prediction Based on Daily Behavioral Features. (in preparation) ← <u>This journal paper is an extension of the UbiComp submission and will be submitted to a proper journal after the UbiComp submission is finally accepted.</u>